FSE PROGRAM SUMMARY DOCUMENT #2.02

PREDIK

R.E.McFarland

Marcus Phipps

March 1991

National Aeronautics and
Space Administration

Document: FSE PROGRAM SUMMARY #2, March 21, 1991

Name: PREDIK, version 1.5

Author: R.E. McFarland, NASA

Documentation: Marcus Phipps

Function: This program is for use by Computer-Generated Imagery (CGI) drive programs to predict positions and angles, as required, in order to compensate for transport delay in the mainframe-to-CGI presentation path.

Language: FORTRAN, Computer Portable.

Location:
(1) SIMDEV VAX, $DISK1:[STRIKE]PREDIK.FOR.
(2) FSD VAX, $LIBRARY:[STRIKE]PREDIK.FOR.
(3) ADDEV VAX, $DISK1:[STRIKE]PREDIK.FOR
(4) Author, PC Diskette.

References:

(1) "A Standard Kinematic Model for Flight Simulation at NASA/Ames Research Center", NASA CR-2497, Jan. 1975 (Computer Sciences Corporation).

(2) "CGI Delay Compensation", NASA TM 86703, Jan. 1986.

(3) "Transport Delay Compensation for Computer-Generated Imagery Systems", NASA TM 100084, Jan. 1988

Summary:

Subroutine PREDIK standardizes the operation of the prediction algorithm. It predicts both angles and positions for the CGI drives. Since the CGI processors operate in parallel and the information received flows serially through the processors, there is a time delay between the transmittal of information and scene presentation.

Setup for the program is controlled by three BLOCK COMMON variables: ICOMPN, WTUNE, and THEORY. THEORY is the prediction interval in seconds that determines how far into the future the program will predict. WTUNE is the tuned frequency. ICOMPN must be set to unity to enable predictions. Otherwise the output CGI drive vector will be merely the uncompensated positions (they are properly transformed to "runway coordinates").

The prediction is carried out through several phases. The compensation algorithm extrapolation coefficients C0, C1, and C2 are computed in I.C.

(Initial Condition) mode. Local Frame-to-Body Frame derivatives, XPRD, YPRD, ZPRD, not found in STRIKE are generally developed (see Appendix C of Reference (1)). These quantites are currently used only by DIG1 software, although CT5A software may also require them in the future.

The velocity ladder (array CGIV with six rows of three values) contains the present, and two past values of the six (positional and rotational) rates. These values are set equal in I.C. mode. When the model goes into operate mode, the velocity values use a "ladder-down" scheme, including a "smoothed starting gate."

Reference (3) should be consulted for definitions. A brief explanation follows: For the DIG1 system, a value of about 0.0917 seconds for THEORY is appropriate, due to the fact that the three processors of this system each operate at 30 HZ. For the CT5A system, with four processors each operating at 50 HZ, the value for THEORY is appropriately 0.075 seconds. The value for WTUNE is required in rad/sec. As shown in Ref. (3), this value should be the equivalent of about 3 HZ. (18.85 rad/sec).

ICOMPN is the flag that enables PREDIK's prediction logic. It may be turned on and off at will, in all modes.

Velocity data is also made available from subroutine PREDIK. This data is currently used only by the DIG1 system, wherein it is utilized by a projection (or smoothing) process within the first pipeline processor. The purpose of the "smoothing operation" is to synchronize asynchronous signals between the mainframe and CGI computer system. The operation extrapolates signals over small and variable intervals. See Ref. (2).

Using the current, past and previous values of velocity along with the current position values, and by using the coefficients computed in the algorithm development, a prediction can be made as to the positions and angles -THEORY- seconds later. This operation is standardized in subroutine PREDIK, wherein all outputs are in "runway coordinates."

Communications with BASIC COMMON

In order to communicate with COMMON arrays standardized for simulation models (BASIC), the following equivalences appear within subroutine PREDIK.

Inputs:

Name	Array	Description (inputs)
PHIR	A(4)	Roll Euler angle in local frame (rad)
THETR	A(5)	Pitch Euler angle in local frame (rad)
PSIR	A(6)	Yaw Euler angle in local frame (rad)
PHID	A(7)	Roll Euler rate in local frame (rad/sec)
THED	A(8)	Pitch Euler rate in local frame (rad/sec)
PSID	A(9)	Yaw Euler rate in local frame (rad/sec)
T11	A(16)	COS(THETR) * COS(PSIR)
T21	A(17)	SIN(PHIR) * SIN(THETR) * SIN(PSIR) - COS(PHIR) * SIN(PSIR)
T31	A(18)	COS(PHIR) * SIN(THETR) * COS(PSIR) + SIN(PHIR) * SIN(PSIR)
T12	A(19)	COS(THETR) * SIN(PSIR)
T22	A(20)	SIN(PHIR) * SIN(THETR) * SIN(PSIR) + COS(PHIR) * COS(PSIR)
T32	A(21)	COS(PHIR) * SIN(THETR) * SIN(PSIR) - SIN(PHIR) * COS(PSIR)
T13	A(22)	-SIN(THETR)
T23	A(23)	SIN(PHIR) * COS(THETR)
T33	A(24)	COS(PHIR) * COS(THETR)
PT	A(46)	Total roll rate of the Local-Frame (rad/sec)
QT	A(47)	Total pitch rate of the Local-Frame (rad/sec)
RT	A(48)	Total yaw rate of the Local-Frame (rad/sec)
ALTD	A(80)	Rate of change of altitude (ft/sec)
XLOND	A(81)	Rate of change of longitude (rad/sec)
XLATD	A(82)	Rate of change of latitude (rad/sec)
XPR	A(103)	Distance of pilot eye position down the runway, Runway-Frame (ft)
YPR	A(104)	Distance of pilot eye position to the right of the runway, Runway-Frame (ft)

HPR	A(105)	Height of the pilot eye position above the runway, Runway-Frame (ft)
RR	A(108)	RR = (RE) radius of the earth + (HR) height of runway (ft)
CLATR	A(113)	Cosine of the runway latitude (XLATR)
STHETR	A(114)	Sine of the runway angle (THETRR)
CTHETR	A(115)	Cosine of the runway angle (THETRR)
DT2	A(168)	Second loop frame time (sec)
XP	A(171)	X position of pilot w/r/t C.G. (ft)
YP	A(172)	Y position of pilot w/r/t C.G. (ft)
ZP	A(173)	Z position of pilot w/r/t C.G. (ft)
ZPE	A(492)	Z Distance from eye point to motion point (+ft)

The following are control inputs to PREDIK.

Name	Array	Description
WTUNE	A(488)	The tuned frequency in RAD/SEC. This parameter is set in BLOCK COMMON and should normally be about 18.85 rad/sec (3 HZ.)
THEORY	A(489)	The theoretical transport delay of the CGI system in seconds. The prediction interval should be about 0.0917 for the DIG. or about 0.075 for the CT5A.
IMODE	IA(1)	Mode control integer where (-1) = I.C.; (0) = HOLD; and (+1) = OPERATE.
ICOMPN	IA(248)	The enable flag for CGI visual delay compensation. The values for ICOMPN are either 0 or 1. If ICOMPN is not set, the output vector will simply consist of the values XPR, YPR, HPR, PHIR, THETR, and PSIR.

Outputs

Subroutine PREDIK's outputs are a specially created COMMON buffer (vector) containing XCGI, YCGI, ZCGI, PHICGI, THTCGI and PSICGI (see descriptions below), beginning in the location A(447). The positional elements of this vector emulate the pilot's position with respect to the runway. The linear derivatives of these quantities, as required by the compensation scheme, are computed in PREDIK.

When enabled (ICOMPN=1), the output vector contains the predicted positions for the transmittal to a CGI drive routine. Associated velocities are also included in this buffer. They are XPRD, YPRD, ZPRD, PHDCGI, THDCGI, and PSDCGI. The first three derivatives are actually computed in PREDIK (because they are not provided by STRIKE or SMART). The last three values are provided elsewhere in the BASIC system, but are replicated within PREDIK for the convenience of the CGI driving subroutine.

Name	Array	Description (outputs)
XCGI	A(447)	X Drive for CGI (ft)
YCGI	A(448)	Y Drive for CGI (ft)
ZCGI	A(449)	Z Drive for CGI (ft)
PHICGI	A(450)	Roll Drive of CGI (rad)
THTCGI	A(451)	Pitch Drive of CGI (rad)
PSICGI	A(452)	Yaw Drive of CGI (rad)
XPRD	A(453)	X velocity drive for CGI (computed in PREDIK)
YPRD	A(454)	Y velocity drive for CGI (computed in PREDIK)
ZPRD	A(455)	Z velocity drive for CGI (computed in PREDIK)
PHDCGI	A(456)	Roll velocity drive for CGI (rad/sec)
THDCGI	A(457)	Pitch velocity drive for CGI (rad/sec)
PSDCGI	A(458)	Yaw velocity drive for CGI (rad/sec)

```
C PREDIK.FOR
C      *****************************************************
C      * STRIKE POSITION PREDICTION ROUTINE FOR CGI DRIVES *
C      *****************************************************
C
C *************************************************************
C * FOR USE BY CGI DRIVE ROUTINES TO PREDICT POSITIONS AND ANGLES *
C * IN THE RUNWAY FRAME.  VELOCITIES PROVIDED BY STRIKE.          *
C *************************************************************
C
      SUBROUTINE PREDIK
C
C ----------------------------------------------------------------------
C                     CREATION AND MODIFICATION
C ----------------------------------------------------------------------
C VERSION 1.0 - MARCH 16, 1988 R. E. MCFARLAND  - NASA -
C VERSION 1.1 - MARCH 28, 1988, ASSIGNED OUTPUT VECTOR
C VERSION 1.3 - DEC. 14, 1989, ASSIGNED OUTPUT VELOCITY VECTOR
C VERSION 1.4 - FEB. 6, 1990, CHANGED SENSE OF ZCGI AND DERIVATIVE.
C                             ALSO PUT IN ZPE LOGIC.
C VERSION 1.5 - JAN. 31, 1991, UPDATED COMMENTS CONCERNING VALUES FOR
C                              THEORY, AND ESTABLISHED RATE NAMES.
C VERSION 1.6 - FEB. 13, 1991, PURGED OF INTERNAL DERIVATIVES.  SINCE
C                              THESE COMPUTATIONS HAVE BEEN MOVED TO
C                              STRIKE, THIS PROGRAM IS NO LONGER
C                              COMPATIBLE WITH SMART...
C
C ----------------------------------------------------------------------
C                     SIGNIFICANT VARIABLES
C ----------------------------------------------------------------------
C ----------------------------------------------------------------------
C                            INPUTS
C ----------------------------------------------------------------------
C
C  VARIABLES ARRAY LOCATION & DEFINITION                 UNIT    SOURCE
C
C  IMODE    IA(  1)  =-: I.C., =0: HOLD, =+: OPERATE.
C  ICOMPN   IA(248)  SWITCH.  1=COMPENSATE, 0=DON'T
C
C  XPR       A(103)  DIST. OF PILOT DOWN RUNWAY         FT      STRIKE
C  YPR       A(104)  DIST. OF PILOT RIGHT OF RUNWAY     FT      ..
C  HPR       A(105)  DIST. OF PILOT ABOVE RUNWAY        FT      ..
C  PHIR      A(  4)  ROLL ANGLE  L-FRAME                RAD     STRIKE
C  THETR     A(  5)  PITCH                              ..      ..
C  PSIR      A(  6)  YAW                                ..      ..
C
C  THE FOLLOWING ARE OUTPUTS OF STRIKE, NOT OF THIS PROGRAM.
C  THEY ARE INPUTS TO THIS PROGRAM (AND SMART DOES NOT COMPUTE THEM).
C
C  XPRD      A(453)   PILOT POSITIONAL RATES   FT/SEC
C  YPRD      A(454)   ..
C  ZPRD      A(455)   ..
C  PHDCGI    A(456)  PILOT ANGULAR RATES                RAD/SEC
C  THDCGI    A(457)   ..
C  PSDCGI    A(458)   ..
C
C  DT2       A(168)  CYCLE TIME                         SEC     DATA
C  WTUNE     A(488)  EQUIVALENT TO ABOUT 3 HZ.          R/S     ..
C  THEORY    A(489)  VARIES WITH SYSTEM.  SEE NOTES.    SEC     ..
C
C ----------------------------------------------------------------------
C                    OUTPUTS   (DESTINATION:  BVISUAL)
C ----------------------------------------------------------------------
C
C            LINEAR AND ANGULAR POSITIONS ARE PREDICTED
C            THEORY  SECONDS INTO THE FUTURE.
C
C  XCGI      A(447)   PREDICTED PILOT POSITIONS         FT
C  YCGI      A(448)   ..
C  ZCGI      A(449)   ..
C  PHICGI    A(450)   PREDICTED PILOT ANGLES            RAD
C  THTCGI    A(451)   ..
```

```
C  PSICGI   A(452)    ..
C
C                    PREDICTED POSITIONS
C (1) XCGI     -  PILOT X POSITION W/R/T RUNWAY (FT)   (CGIDR(1))
C (2) YCGI     -  PILOT Y POSITION W/R/T RUNWAY (FT)
C (3) ZCGI     -  PILOT Z POSITION W/R/T RUNWAY (FT)  ( + DOWN)
C (4) PHICGI   -  ROLL ATTITUDE (RAD)
C (5) THTCGI   -  PITCH ATTITUDE (RAD)
C (6) PSICGI   -  YAW ATTITUDE (RAD)
C
C THE ABOVE SIX-VECTOR OCCUPIES A(447) TO A(452).  IN CELLS A(453) TO
C A(458) THE RATES OF CHANGE OF THESE QUANTITIES APPEARS - AS COMPUTED
C BY STRIKE.  THESE ARE THE RATES THAT SHOULD BE SENT TO THE CGI.
C (THESE RATES ARE NOT PREDICTED).  CURRENTLY, ONLY THE DIG SYSTEM
C USES RATES.  THE CT5A SYSTEM, IN ITS ATTEMPT TO ACCOMMODATE
C ASYNCHRONOUS DELAY, USES DIFFERENCES IN POSITIONS.
C
C ----------------------------------------------------------------------
C                              LOCAL
C ----------------------------------------------------------------------
C
C  CGIV(6,3)   CURRENT AND TWO PAST VALUES OF 6 VELOCITES.
C
C  KSTART      USED INTERNALLY TO AVOID JUMP UPON ENTERING OPERATE MODE.
C
C  C0,C1,C2    THE COMPENSATION COEFFICIENTS.  THESE ARE FUNCTIONS OF
C              WTUNE, THEORY AND DT2 (SEE REFERENCES).
C
C ----------------------------------------------------------------------
C                              COMMONS
C ----------------------------------------------------------------------
C
      COMMON/XFLOAT/A(500)/IFIXED/IA(250)
C
C ----------------------------------------------------------------------
C                              EQUIVALENCES
C ----------------------------------------------------------------------
C
      EQUIVALENCE (A(  4), PHIR )
      EQUIVALENCE (A(  5), THETR )
      EQUIVALENCE (A(  6), PSIR )
      EQUIVALENCE (A(103), XPR )
      EQUIVALENCE (A(104), YPR )
      EQUIVALENCE (A(105), HPR )
      EQUIVALENCE (A(168), DT2 )
C
C THIS 6-VECTOR IS THE OUTPUT CGI DRIVE VECTOR, ASSIGNED 3/28/88
      EQUIVALENCE (A(447), XCGI, CGIDR(1))
      EQUIVALENCE (A(448), YCGI)
      EQUIVALENCE (A(449), ZCGI)
      EQUIVALENCE (A(450), PHICGI)
      EQUIVALENCE (A(451), THTCGI)
      EQUIVALENCE (A(452), PSICGI)
C
C VELOCITIES ARE NOT 'PREDICTED'.  THIS IS IMPORTANT.  ONLY THE
C CURRENT VALUE OF VELOCITIES ARE USED IN THE DIG SYSTEM, AND THESE
C VALUES ARE TO 'PROJECT' THROUGH THE ASYMMETRIC DELAY CAUSED BY
C THE DOUBLE-BUFFER SCHEME IN THE FIRST PIPELINE PROCESSOR.
C THE ABOVE STATEMENT CONCERNS ONLY THE DIG SYSTEM.  PERHAPS LATER
C THE CT5A SYSTEM WILL ALSO USE THESE VELOCITY VALUES.
C
C THE BVISUAL PROGRAM SHOULD USE THE FOLLOWING VELOCITIES FOR
C TRANSMITTAL TO CGI (DIG).
C THESE ARE COMPUTED IN STRIKE (NOT SMART):
      EQUIVALENCE (A(453), XPRD )
      EQUIVALENCE (A(454), YPRD )
      EQUIVALENCE (A(455), ZPRD )
      EQUIVALENCE (A(456), PHDCGI)
      EQUIVALENCE (A(457), THDCGI)
      EQUIVALENCE (A(458), PSDCGI)
C
C NOTE THAT THE OUTPUTS ARE A LONG, CONVENIENT VECTOR, FOR USE IN
```

```
C THE VISUAL DRIVE ROUTINE.  PREDIK SHOULD BE CALLED BEFORE THE
C CALL TO A VISUAL DRIVE ROUTINE.                     ------
C THE VECTOR IS 12 LONG.
C
C CONTROL INPUTS TO PREDIK ARE ICOMPN, WTUNE AND THEORY.
C
C ICOMPN = 1 TO ENABLE PREDICTIONS.  IF ICOMPN IS NOT SET THE
C            OUTPUT VECTOR (ABOVE) WILL SIMPLY CONSIST OF THE
C            NON-PREDICTED VALUES (HENCE, GENERAL APPLICABILITY).
C
C WTUNE      THE TUNED FREQUENCY IN RAD/SEC.  THIS IS SET IN BLOCK
C            COMMON AND SHOUD BE ABOUT 18.8 RAD/SEC (3 HZ).
C
C THEORY     THE PREDICTION INTERVAL IN SECONDS.  THIS SHOULD BE ABOUT:
C
C            THEORY = 0.0917  FOR THE D.I.G. SYSTEM (SEC)
C            THEORY = 0.0750  FOR THE CT5A SYSTEM (SEC)
C
C THESE NUMBERS COME ABOUT FROM:
C
C      THEORY = (NUMBER OF PROCESSORS - 1/4)*(CYCLE TIME OF A PROCESSOR)
C
C
C A REMINDER:  THE STRIKE SYSTEM NEVER SETS COMMON VALUES WITHIN
C              INDIVIDUAL ROUTINES.  THEY ARE ONLY SET WITHIN A
C              BLOCK DATA PROGRAM, OR RAD FILES, OR MANUALLY, IF YOU
C              LIKE TO TYPE...
C
C THIS PROGRAM IS NOT COMPATIBLE WITH -SMART- BECAUSE SMART DOES NOT
C COMPUTE THE REQUIRED DERIVATIVES.
C
      EQUIVALENCE (A(488), WTUNE)
      EQUIVALENCE (A(489), THEORY)
      EQUIVALENCE (IA(  1), IMODE)
      EQUIVALENCE (IA(248), ICOMPN)
C
C ---------------------------------------------------------------------
C                        DECLARATIONS
C ---------------------------------------------------------------------
C
      DIMENSION CGIDR(6),CGIV(6,3)
C
C ---------------------------------------------------------------------
C                        NO DATA
C ---------------------------------------------------------------------
C                        REFERENCES
C ---------------------------------------------------------------------
C
C NASA TM 100084, JAN. 1988 BY R. E. MCFARLAND:
C TRANSPORT DELAY COMPENSATION FOR COMPUTER-GENERATED IMAGERY SYSTEMS
C
C NASA TM 86703, JAN. 1986 BY R. E. MCFARLAND:
C CGI DELAY COMPENSATION
C
C NASA CR 2497, JAN. 1975 BY R. E. MCFARLAND:
C A STANDARD KINEMATIC MODEL FOR FLIGHT SIMULATION AT NASA/AMES
C RESEARCH CENTER
C
C FSE PROGRAM SUMMARY #2.01, JAN. 1991, BY MCFARLAND AND PHIPPS
C
C ---------------------------------------------------------------------
C                        EXECUTABLE CODE
C ---------------------------------------------------------------------
C
      IF(IMODE.GT.0) GO TO 30
      IF(IMODE.EQ.0) RETURN
C
C ---------------------------------------------------------------------
C                        I.C. MODE
C ---------------------------------------------------------------------
C
      IF((WTUNE*THEORY).LE.0.0) THEN
```

```
          C0 = 0.0
          C1 = 0.0
          C2 = 0.0
          GO TO 10
        END IF
C
C COMPUTE CGI COMPENSATION PARAMETERS
        THEX = WTUNE*DT2
        PSIX = WTUNE*THEORY
        CT   = COS(THEX)
        ST   = SIN(THEX)
        CP   = COS(PSIX)
        SP   = SIN(PSIX)
        OMCT = 1.0 - CT
        DEN  = 1.0/(2.0*WTUNE*ST*OMCT)
        C0   = DEN*(ST*(PSIX + SP*(1.0 - 2.0*CT))
     1         + (0.5*THEX*ST - CP*OMCT)*(1.0 + 2.0*CT))
        C1   = DEN*ST*( 2.0*(ST*CP + CT*SP)
     1         - 2.0*PSIX*CT - THEX*(1.0 + CT) )
        C2   = DEN*(ST*(PSIX - SP + 0.5*THEX) - CP*OMCT)
C
  10    CGIDR(1) = XPR
        CGIDR(2) = YPR
        CGIDR(3) = - HPR
        CGIDR(4) = PHIR
        CGIDR(5) = THETR
        CGIDR(6) = PSIR
C
C INITIALIZE VELOCITY LADDER, BUT DO NOT PREDICT
C POSITIONS (USING THESE VELOCITIES) IN I.C. MODE
        CGIV(1,1) = XPRD
        CGIV(2,1) = YPRD
        CGIV(3,1) = ZPRD
        CGIV(4,1) = PHDCGI
        CGIV(5,1) = THDCGI
        CGIV(6,1) = PSDCGI
C
        DO 20 J=2,3
        DO 20 I=1,6
  20    CGIV(I,J) = CGIV(I,1)
C
        KSTART = 1
        RETURN
C
C ---------------------------------------------------------------------
C                             OPERATE MODE
C ---------------------------------------------------------------------
  30    CONTINUE
C
C LADDER DOWN VELOCITY VALUES
        DO 40 I=1,6
        CGIV(I,3) = CGIV(I,2)
  40    CGIV(I,2) = CGIV(I,1)
C
        CGIV(1,1) = XPRD
        CGIV(2,1) = YPRD
        CGIV(3,1) = ZPRD
        CGIV(4,1) = PHDCGI
        CGIV(5,1) = THDCGI
        CGIV(6,1) = PSDCGI
C
C PICKUP NEW BASELINE POSITIONAL VALUES
        CGIDR(1) = XPR
        CGIDR(2) = YPR
        CGIDR(3) = - HPR
        CGIDR(4) = PHIR
        CGIDR(5) = THETR
        CGIDR(6) = PSIR
C
C CHECK FOR OPERATE MODE TURN-OFF OF ICOMPN
        IF(ICOMPN.EQ.1) GO TO 50
        KSTART = 1
```

```
      RETURN
   50 CONTINUE
C
      IF(KSTART.EQ.0) GO TO 70
C SMOOTH STARTING GATE.  ONLY ON THE 2ND AND LATER OPERATE-MODE
C PASSES ARE THREE DISTINCT VELOCITY VALUES AVAILABLE.
      KSTART = 0
      DO 60 I = 1,6
   60 CGIV(I,3) = 2.0*CGIV(I,2) - CGIV(I,1)
   70 CONTINUE
C
C PREDICT VALUES -THEORY- SECONDS LATER (ADD APPROPRIATE INCREMENTS)
      DO 80 I=1,6
   80 CGIDR(I) = CGIDR(I)+C0*CGIV(I,1)+C1*CGIV(I,2)+C2*CGIV(I,3)
C
      RETURN
      END
```

```
C PRODIK.FOR
C     ***************************************************************
C     * STRIKE VECTOR POSITION PREDICTION ROUTINE FOR CGI DRIVES *
C     ***************************************************************
C
C ******************************************************************
C * FOR USE BY CGI DRIVE ROUTINES TO PREDICT POSITIONS AND ANGLES *
C * FOR UP TO FIVE OBJECTS.                                        *
C ******************************************************************
C
      SUBROUTINE PRODIK(NPRED,ICOMP,WCUT,PIPE,PVECTC,VVECTC,DR)
C
C ----------------------------------------------------------------------
C                  CREATION AND MODIFICATION
C ----------------------------------------------------------------------
C VERSION 1.0 - FEB. 19, 1991  R. E. MCFARLAND -NASA-
C ----------------------------------------------------------------------
C
C                  SIGNIFICANT VARIABLES
C ----------------------------------------------------------------------
C ----------------------------------------------------------------------
C                         INPUTS
C ----------------------------------------------------------------------
C
C  VARIABLES ARRAY LOCATION & DEFINITION                  UNIT    SOURCE
C
C  IMODE    IA(  1)  =-: I.C., =0: HOLD, =+: OPERATE.
C  DT2       A(168)  CYCLE TIME                           SEC
C
C                    IN CALLING SEQUENCE:
C
C  NPRED             DESIGNATES OBJECT NUMBER FROM 1 TO 5
C  ICOMP             SWITCH.  1=COMPENSATE, 0=DON'T
C  WCUT              EQUIVALENT TO ABOUT 3 HZ.            R/S
C  PIPE              VARIES WITH SYSTEM.  SEE NOTES.      SEC
C  PVECTC            VECTOR OF POSITIONS AND ANGLES       FT & RAD
C  VVECTC            VECTOR OF RATES OF ABOVE             FPS AND RAD/SEC
C
C ----------------------------------------------------------------------
C                    OUTPUTS  (DESTINATION:  BVISUAL)
C ----------------------------------------------------------------------
C
C             LINEAR AND ANGULAR POSITIONS ARE PREDICTED
C             PIPE  SECONDS INTO THE FUTURE.
C
C  DR                6-VECTOR OF PREDICTED POSITIONS AND ANGLES
C
C ----------------------------------------------------------------------
C                         LOCAL
C ----------------------------------------------------------------------
C
C  CGIV(6,3,5)  CURRENT AND TWO PAST VALUES OF 6 VELOCITES, UP TO 5
C               OBJECTS.
C
C  KSTART(5)  USED INTERNALLY TO AVOID JUMP UPON ENTERING OPERATE MODE.
C
C  C0(5)      THE COMPENSATION COEFFICIENTS.  THESE ARE FUNCTIONS OF
C  C1(5)      WCUT, PIPE AND DT2 (SEE REFERENCES).
C  C2(5)
C
C ----------------------------------------------------------------------
C                         COMMONS
C ----------------------------------------------------------------------
C
      COMMON/XFLOAT/A(500)/IFIXED/IA(250)
C
C ----------------------------------------------------------------------
C                         EQUIVALENCES
C ----------------------------------------------------------------------
C
      EQUIVALENCE (A(168), DT2  )
      EQUIVALENCE (IA( 1), IMODE)
```

```
C
C
C VELOCITIES ARE NOT 'PREDICTED'.  THIS IS IMPORTANT.  ONLY THE
C CURRENT VALUE OF VELOCITIES ARE USED IN THE DIG SYSTEM, AND THESE
C VALUES ARE TO 'PROJECT' THROUGH THE ASYMMETRIC DELAY CAUSED BY
C THE DOUBLE-BUFFER SCHEME IN THE FIRST PIPELINE PROCESSOR.
C THE ABOVE STATEMENT CONCERNS ONLY THE DIG SYSTEM.  PERHAPS LATER
C THE CT5A SYSTEM WILL ALSO USE THESE VELOCITY VALUES.
C
C CONTROL INPUTS TO PRODIK ARE ICOMP, WCUT AND PIPE.
C
C ICOMP   = 1 TO ENABLE PREDICTIONS FOR OBJECT NUMBER NPRED.
C             IF ICOMP IS NOT SET THE
C             OUTPUT VECTOR (ABOVE) WILL SIMPLY CONSIST OF THE
C             NON-PREDICTED VALUES (HENCE, GENERAL APPLICABILITY).
C
C WCUT        THE TUNED FREQUENCY IN RAD/SEC.  THIS
C             SHOULD BE ABOUT 18.8 RAD/SEC (3 HZ).
C
C PIPE        THE PREDICTION INTERVAL IN SECONDS.  THIS SHOULD BE ABOUT:
C
C             PIPE   = 0.0917  FOR THE D.I.G. SYSTEM (SEC)
C             PIPE   = 0.0750  FOR THE CT5A SYSTEM (SEC)
C
C THESE NUMBERS COME ABOUT FROM:
C
C      PIPE   = (NUMBER OF PROCESSORS - 1/4)*(CYCLE TIME OF A PROCESSOR)
C
C
C A REMINDER:  THE STRIKE SYSTEM NEVER SETS COMMON VALUES WITHIN
C              INDIVIDUAL ROUTINES.  THEY ARE ONLY SET WITHIN A
C              BLOCK DATA PROGRAM, OR RAD FILES, OR MANUALLY, IF YOU
C              LIKE TO TYPE...
C
C THIS PROGRAM IS NOT COMPATIBLE WITH -SMART- BECAUSE SMART DOES NOT
C COMPUTE THE REQUIRED DERIVATIVES FOR OBJECT #1 (OWNSHIP).
C
C
C ---------------------------------------------------------------------
C                            DECLARATIONS
C ---------------------------------------------------------------------
C
      DIMENSION PVECTC(6),VVECTC(6),DR(6),CGIV(6,3,5)
      DIMENSION C0(5),C1(5),C2(5),KSTART(5)
C
C ---------------------------------------------------------------------
C                             NO DATA
C ---------------------------------------------------------------------
C                            REFERENCES
C ---------------------------------------------------------------------
C
C NASA TM 100084, JAN. 1988 BY R. E. MCFARLAND:
C TRANSPORT DELAY COMPENSATION FOR COMPUTER-GENERATED IMAGERY SYSTEMS
C
C NASA TM 86703, JAN. 1986 BY R. E. MCFARLAND:
C CGI DELAY COMPENSATION
C
C FSE PROGRAM SUMMARY #2.01, JAN. 1991, BY MCFARLAND AND PHIPPS
C
C ---------------------------------------------------------------------
C                          EXECUTABLE CODE
C ---------------------------------------------------------------------
C
      IF(IMODE.GT.0) GO TO 130
      IF(IMODE.EQ.0) RETURN
C
C ---------------------------------------------------------------------
C                             I.C. MODE
C ---------------------------------------------------------------------
C
      IF((WCUT*PIPE).LE.0.0) THEN
        C0(NPRED) = 0.0
```

```
         C1(NPRED) = 0.0
         C2(NPRED) = 0.0
         GO TO 70
      END IF
C
C COMPUTE CGI COMPENSATION PARAMETERS
      THEX = WCUT*DT2
      PSIX = WCUT*PIPE
      CT   = COS(THEX)
      ST   = SIN(THEX)
      CP   = COS(PSIX)
      SP   = SIN(PSIX)
      OMCT = 1.0 - CT
      DEN  = 1.0/(2.0*WCUT*ST*OMCT)
      C0(NPRED)    = DEN*(ST*(PSIX + SP*(1.0 - 2.0*CT))
     1          + (0.5*THEX*ST - CP*OMCT)*(1.0 + 2.0*CT))
      C1(NPRED)    = DEN*ST*( 2.0*(ST*CP + CT*SP)
     1          - 2.0*PSIX*CT - THEX*(1.0 + CT) )
      C2(NPRED)    = DEN*(ST*(PSIX - SP + 0.5*THEX) - CP*OMCT)
C
  70  CONTINUE
      DO 80 I=1,6
  80  DR(I) = PVECTC(I)
C
C INITIALIZE VELOCITY LADDER, BUT DO NOT PREDICT
C POSITIONS (USING THESE VELOCITIES) IN I.C. MODE
C
      DO 90 I=1,6
  90  CGIV(I,1,NPRED) = VVECTC(I)
C
      DO 100 J=2,3
      DO 100 I=1,6
 100  CGIV(I,J,NPRED) = VVECTC(I)
C
      KSTART(NPRED) = 1
      RETURN
C
C ---------------------------------------------------------------------
C                        OPERATE MODE
C ---------------------------------------------------------------------
 130  CONTINUE
C
C LADDER DOWN VELOCITY VALUES (NEED CURRENT AND 2 PAST VALUES)
      DO 140 I=1,6
      CGIV(I,3,NPRED) = CGIV(I,2,NPRED)
 140  CGIV(I,2,NPRED) = CGIV(I,1,NPRED)
C
      DO 150 I=1,6
 150  CGIV(I,1,NPRED) = VVECTC(I)
C
C PICKUP NEW BASELINE POSITIONAL VALUES
C
      DO 160 I=1,6
 160  DR(I) = PVECTC(I)
C
C CHECK FOR OPERATE MODE TURN-OFF OF ICOMP
      IF(ICOMP.EQ.1) GO TO 170
      KSTART(NPRED) = 1
      RETURN
 170  CONTINUE
C
      IF(KSTART(NPRED).EQ.0) GO TO 190
C SMOOTH STARTING GATE.  ONLY ON THE 2ND AND LATER OPERATE-MODE
C PASSES ARE THREE DISTINCT VELOCITY VALUES AVAILABLE.
      KSTART(NPRED) = 0
      DO 180 I = 1,6
 180  CGIV(I,3,NPRED) = 2.0*CGIV(I,2,NPRED) - CGIV(I,1,NPRED)
C
 190  CONTINUE
C
C PREDICT VALUES -PIPE- SECONDS LATER (ADD APPROPRIATE INCREMENTS)
      DO 200 I=1,6
```

```
  200   DR(I) = DR(I) + C0(NPRED)*CGIV(I,1,NPRED)
     1          + C1(NPRED)*CGIV(I,2,NPRED) + C2(NPRED)*CGIV(I,3,NPRED)
C
        RETURN
        END
```